Final Scientific and Technical Report

Phase II Program

Project Title

Parametric Crowd Generation Software for MS&T Simulations and Training

Issued by U.S. Army Aviation and Missile Command
Under Contract Number: W31P4Q-05-C-0286

Topic Number: D2-0421
ARPA Order: U708-60
Agency: DARPA

Sponsored by
Defense Advanced Research Projects Agency (DoD)
(Controlling DARPA Office)

Contractor

Online Alchemy
8000 Anderson Square
Suite 110
Austin, TX 78757

Phone: 001-512-419-9001
http://www.onlinealchemy.com

Approved for public release; distribution unlimited.

Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
20 FEB 2007	**N/A**	**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Parametric Crowd Generation Software for MS&T Simulations and Training	**W31P4Q-05-C-0286**
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Online Alchemy**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**SAR**	**20**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

SB041-009 D2-0421 Final Report
Table of Contents

Please direct any comments or questions about this Final Report to:

Craig Fryar
Online Alchemy
craig@onlinealchemy.com
+001-512-419-9001 mobile

Application of Online Alchemy's Dynemotion™ Technology to Parametric Crowd Generation for use in MS&T simulations and training

Executive Summary

Online Alchemy, Inc. ("Company") develops software and systems focused on the creation of more realistic computer generated characters for use in simulation software. Their primary business is developing online video game entertainment products, known as massively multiplayer online games, or "MMOGs". Current MMOG offerings lack realistic non-player computer generated characters ("NPCs") and instead typical NPCs are one-dimensional "quest vending machines" that basically assist player characters by dispending missions and providing information needed to progress in the game.

In 2002 the Company's founder embarked on an effort to develop an NPC engine, based on accepted principles of human psychology, behavioral models and interactions, that would extend capabilities well beyond existing artificial intelligence (mostly focused on path finding and movement) into new areas of artificial psychology. An early version of this engine was prototyped in 2003-04 at which time the Company filed for intellectual property protection of the various innovations developed as part of the Dynemotion™ engine. In essence, this engine acts to empower the "head and heart" of NPCs. It is designed to operate in conjunction with existing AI technologies, which are mainly concerned with NPC movement, that is, from the "neck down".

Shortly after this time the Company proposed and was awarded a Phase I SBIR grant on 2004MAY05 entitled "IQ for Avatars" which included a definition and proposed design for a project to develop a simulation environment to test for expected and unexpected behaviors rendered by this engine. The result was the 2005FEB22 award of a six month Phase III SBIR contract under which this proposed design for a "House Search" simulation was implemented and delivered. This "fish-bowl" non-interactive "House Search" simulation, based on actual in-theater SME input, proved successful in demonstrating Dynemotion's early abilities as a People Engine™ in which two sets of parametrically generated groups (Arabic and Military) were auto-created and set about an emergent behavior scenario. Outcomes of simulation runs resulted in highly believable interactions, indistinguishable from traditionally scripted simulation scenarios.

The main distinction was that each character was moving, observing, acting, interacting, reacting based on their own contextual awareness which was in turn affected by individual personalities, goals, relational bias, and in the case of military NPCs, training level. House Search simulation allowed end-users to change NPC generation parameters, after which the simulation could be re-run and outcomes observed. Interestingly these parameter changes would result in sometimes subtle, sometimes dramatic changes in the simulation scenario and outcomes, however, all the while the NPCs performed within expected behavioral norms. The final deliverable for this Phase III project was a demonstration CD which included the "House Search" scenario and documentation for IBM-PC, Windows XP OS multimedia systems.

Following this Phase III SBIR award DARPA granted an additional Phase II award to further the research and development of Dynemotion as a parametric People Engine. This six month project commenced in 2005SEP15 and resulted in the development of a "Parametric Crowd Generator" utility that enabled the definition of groups and crowds of people, including their relationship and interrelations to other groups. These groups were used to populate an Arabic village with townspeople and military personnel, which then performed basic interactions and movements, based on each NPC's contextual awareness of their environment and NPC characters and player characters (PCs).

The outcomes of this project were presented to DARPA/DARWARS and to Total Immersion Software, Inc. ("TIS"), prime contractor for the "Real World" simulation toolbox. As a result the Company was awarded an Option I for this Phase II on 2006JUL02, based upon part 2 of the original Phase II Technical proposal SOW, and is the subject of this final report. The primary objective was delivery of Dynemotion 1.0 API to Total Immersion Software. The scope, effort and deliverables involved are detailed in the following pages.

Keywords: Software, Simulation, Crowd Generation, Non-Player Characters (NPCs), Intelligent Agents

Online Alchemy, Inc., the prime contractor ("Company") for this Phase II, Option I contact award, has completed and fulfilled a revised version of the Tasks proposed in Part 2 of the Statement of Work ("SOW") found in the Technical Proposal for SB041-009 D2-0421. Task revisions were mutually agreed upon by Sponsor and TIS. In the distribution of this report it has now completed and delivered version 1.0 of its Dynemotion™ People Engine™ API ("Dynemotion"), including a full electronic documentation set (in both Adobe Acrobat PDF and Doxygen html formats), sample code, example files and computer videos demonstrating of the technology in a military simulation run-time setting.

The Dynemotion 1.0 API and documentation was delivered December 15, 2006, on schedule, to Total Immersion Software. This was followed by debriefing meetings in January 2007 with their Austin-based development team lead to discuss both the technology and potential timeline for integration of the Dynemotion API into their software simulation client. This is anticipated to occur in late 2007/early 2008.

This project also culminated in the first commercial license of the Dynemotion engine and API. The license was sold to Total Immersion Software for use in a to-be-announced future commercial entertainment simulation project. The license included the software API, documentation and tech support. As a primary goal of the DARPA SBIR program is to foster and facilitate the commercial success of its participants, we are pleased to report this development and to serve as an exemplar of this objective. Our gratitude and thanks to our Sponsor, Dr. Ralph Chatham and also to Ms. Connie Jacobs, Dan Kaufman and the executives at Total Immersion for their contribution to our success in this project.

The Company has also begun discussions with other potential licensees for the Dynemotion technology and intends to focus efforts in 2007-08 on implementing the engine into a next-generation MMOG offering in order to further refine and provide proof-source example of the engine in action. This will likely drive additional sales of licenses for the engine to third-party software and simulation developers.

The Company is now able to demonstrate that the technology, as originally proposed and designed, meets or exceeds the functional specifications. This accomplishment represents a significant breakthrough in the development of computer generated intelligent "avatars", "agents" or non player characters ("NPCs"). The potential application and impact to the DoD includes the ability of this "People Engine" to generated life-like, autonomous, believable NPCs for use in military simulation and training applications. Other potential uses include homeland security preparedness training and use in the commercial world of computer video gaming.

During the execution of this award the company also developed a derivative use of the engine related to reputation management. The engine enables NPCs to share information, including values, culture and reputation of other NPCs and PCs. The Company has created an early prototype of a reputation system based on this ability and has demonstrated this in the context of gaming and non-video game applications. This unanticipated development represents yet another beneficial outcome of this project.

Option I Project Milestones and Timeline

People Engine Prototype	July 2006	Aug 2006	Sept 2006	Oct 2006	Nov 2006	Dec 2006
Milestones & Timeline						
API design						
Sim to Brain API implementation						
Brain to Sim API implementation						
API glue layer						
Architectural refinement						
Cultural behaviors/gestures						
Crowd models/animations						
Scenario/environment art						
QA, refinement, testing, docs						
Dynemotion 1.0 API Deliverable						12/15/06

Dynemotion Technology Overview
Online Alchemy's Dynemotion People Engine technology enables the fast and easy creation of a wide variety of autonomous agents in a simulation program. These agents may be as simple or detailed as desired by the simulation programmer, and are provided with attributes and behaviors that make for highly believable interactions between agents and between agents and human driven avatars. Dynemotion is a C++ linkable library that can be easily integrated with most simulations.

Each Dynemotion agent has his or her own personality, motivational set, emotions, memories, relationships, perceptions, and behaviors. The default personality model is based on the Five Factor model, though this may be easily replaced or extended by the user for some or all agent types by the use of the brain creation tool. The agents' default motivational set is derived from behavioral and neuroscience research, and, like the personalities, may be changed or extended to create agents with more nuanced motivations and responses. Each agent also has a wide variety of emotional responses that affect their interactions, behaviors, memories, and relationships.

Agents in Dynemotion learn from not only their own experiences, but from what they see and are told by other agents. Each maintains a set of emotionally based associations and memories with people, events, and things in their world derived from their experiences, and build relationships and opinions on that basis. Finally, these agents are able to act in the world by executing scripted modular behaviors written in Python. They select their goals and actions based on their current motivations, prior experience, and emotional attachments. The set of behaviors available to the agents is determined by the programmer, and agents can learn new behaviors within the simulation. They also use a form of backward chaining to move to a desired goal state from their current state.

When linked into a simulator and after agents have been created, Dynemotion functions by enabling each agent to think, act, and learn. These cycles happen independently (by default at 250 msec, 1 second, and 3 second intervals) and may have their timing modified easily. During the 'think' part of the cycle each agent perceives objects, people, and other actions in their surroundings and adjusts their motivations and emotions. Notably, this includes fast changes in their displayed disposition and facial expressions. During the 'act' phase the agent uses their current state to decide on a goal and current action, and then attempts to carry out that action. During the 'learn' part of processing, the agent learns from their experiences and processes memories.

Simulation NPC Interface Prototype Including Emotional Space GUI

Dynemotion is not state-based or built on any traditional AI architecture. It has elements of a beliefs-desires-intentions (BDI) architecture and other layered architectures. Neural nets, self organizing maps, and genetic algorithms are not explicitly used in Dynemotion, but it borrows elements from both. Dynemotion also relies on software implementations of psychological and neuroscience models. It uses those such as the Five Factor model of personality and derives from some elements of cognitive, Maslovian, and even post-Freudian psychology. Its internal emotional model is unique, allowing for multiple (often conflicting) emotions within the agent at any time, nuanced affect, and both emotional and propositional reasoning.

Dynemotion System Components
The following is a brief overview of the major system components (Illustrated in Figure 1 below):

Brain – Each thinking entity (single NPC, group of NPCs, etc) must have a brain object. This stores the brain's desire state, memories, associations, and suggestions. On a set interval a brain thinks (modifies its desire state based on its internal state and perception of its surroundings). On another set interval (usually longer than a think cycle) the brain acts (goes through its memories and suggestions and selects the goal which the brain believes will best satisfy its goals). Finally, on a longer cycle, the brain learns (adjusts memory association values based on recent experience).

Dynemotion System Architecture

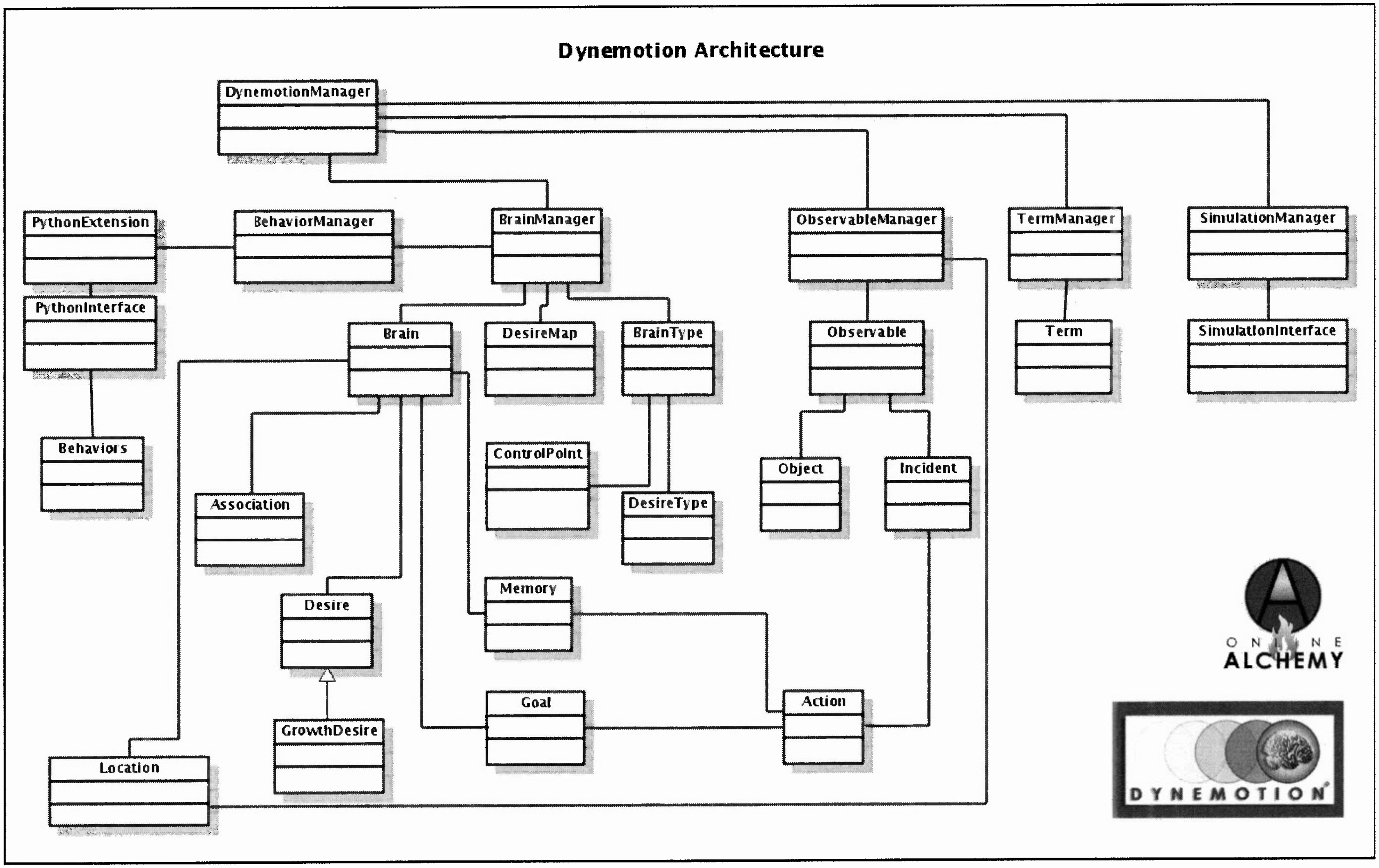


Figure 1 - Major classes and connections in the Dynemotion architecture. The major connection points to the simulation are the SimulationInterface, DynemotionManager, and PythonExtension.

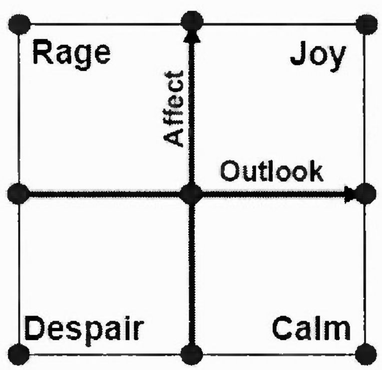
Desire – A function and internal data structures that calculates a motivational factor for the agent. Each desire takes into account its own internal state, personality factors, and recent perceptions to derive a motivational score for the brain. Each desire handles a specific type of motivation such as attraction, hunger, sociability, etc. Desires are defined by their parameters contained in an XML file as part of the brain type definition. Each type of brain defines one or more desires. Together their scores make up the brain's aggregate desire state. The combination of desires in this list tells the brain what its current motivational desires are at any given time; in other words, what it currently needs or wants. The list of desires is determined by the user of Dynemotion for each brain type, but an example of a set of Human desires is included. Desires have values ranging from [0-1.0]. The lower the desire value the more satisfied in that area the brain is. The higher the value the more the brain wishes to act upon, or satisfy, that desire. For example, a high hunger desire will make a person likely to choose 'eat' as an action, while a low hunger desire means the person is not hungry enough to act upon that desire.

Memory – Dynemotion brains remember behaviors they have done in the past, those they have perceived others do, and those they have been told about by others in incident memory. They also remember objects they have perceived in object memory. The behavioral memories are scored based on how they affected the desire state of the brain that performed the remembered action. This allows a brain to use these memories to score the performance of that action based on its current state. For example, a brain is hungry and has a memory of eating food that satisfied its hunger value (the behavior reduced its hunger desire toward zero). In the absence of other desires, the eat goal will be selected when the hunger desire value is high. To a lesser degree, the current state of the performing brain at the time the memory is created is also considered. The object memory allows a brain to remember how it feels about and where it saw a particular object or type of object so that it can go back to that location if the object is needed. Non-physical 'objects' such as skills and values may also be incorporated into object memory.

Associations – Brains store how they feel about things they perceive in associations. Associations refer to a Term, an emotional satisfaction value expressed in terms of the brain's desires, and a strength value. The satisfaction values range from -1.0 to 1.0, with negative numbers being more satisfying (they reduce existing desires). The strength value is an integer indicating how certain or resolute that association is: a low number means the association is weak or tentative; higher values (100+) indicate extensive experience. Associations are used in evaluating observed objects and goals in terms of the agent's existing desires. They change with new experience with the referenced Term, but associations with greater strengths change slowly.

Disposition – The outward view of an agent's current emotional state. This is defined by mapping all desire values to a two-dimensional space using a system of control points that define a value for each desire. The X-axis in the 2D space is Outlook, and the Y axis is Affect (Figure 2, below). Outlook is the relative happiness or emotional valence factor, while Affect is the character's visible emotional energy.

Disposition is most useful for demonstrating emotional values to human participants in the simulation. These values can drive facial expression and bodily postures. In addition, we have found it useful to map the desire values to a color space (Figure 3, below) to provide a qualitative indication of emotion. This color mapping is not strictly necessary and is not part of the Dynemotion engine.



In addition to Outlook and Affect, the Disposition also stores a series of float values corresponding to the weight (or closeness) to all the control points of that brain type. This is used for weighting facial expressions.

Figure 2 - Dynemotion's two dimensional (Outlook, Affect) disposition space. By mapping different desire values to the nine points shown, a continuous map of a nuanced emotional space can be built. The four emotions shown are examples of those at the extremes; many others are also easily modeled.

Figure 3 - This is an image from the simulation developed by Online Alchemy to illustrate the Dynemotion engine for this Option I project. The image depicts a US military character along with several Iraqi civilians. The color sprays show each individual's association with the military character. The colors used correspond to the emotional color wheel (see the discussion of Disposition, above) and the height of the spray indicates the strength of their association. Note that the military character feels 'calmly happy' about himself with a very strong association. One civilian male on the right does not like the character, but has a small color spray indicating little experience with him (a low association strength). Some of the civilians have no color shown, indicating no association with the character at all. These particle effects are just one way to graphically depict associations in the user interface.

Suggestions – A suggestion is an outside pressure affecting the probability of selecting a certain goal. After memories are used to pick goals, the goals are scored, and then any suggestions are applied to those goal scores as well. For example, a brain that scores walk better than greet, could have a greet suggestion that outweighs the scoring difference between walk and greet, causing the brain to greet rather than walk. Suggestions may also be made against an action, reducing its preferability. For example, after an agent has greeted another agent, the greet action can cause a suggestion against greeting the same person again, so that the greeting agent does not continue doing the same action over and over again.

Goal – A goal is an action selected from behavioral memory. A memory is turned into a goal if it is legal for the acting brain's type, and it becomes the agent's current goal if it outscores all others. In other words an animal type brain would never be able to shoot a gun as it's not a legal behavior for that brain type regardless of how many times it saw a human brain perform that action. Goals are scored and then have suggestion modifiers applied to them (see Suggestions, above). The lowest scoring goal that is currently possible is the goal a brain will execute. 'Lowest' score is considered best for implementation reasons. Essentially, the goal that drives current desires closest to zero is the most preferable. If a goal is selected but is not possible on its own, it may cause a 'pursuant' goal to be considered and potentially executed to make the first goal possible. Chains and trees of goals pursuant to some end can be dynamically created this way.

Term – The brain representation of a concept. This can be an object, a verb or a descriptor of an object. Terms are mapped to integers in the system, but those are abstracted out of the users' view in most cases. Terms can also "express" other Terms and can "propose" actions to an observing brain.

Location – Objects, incidents and brains in Dynemotion can all have a location. The location is a 3d point and a named Locale. You can use the named Locale for things like navigation and for referencing in behaviors. The 3d point is used for distance attenuation of incident and object perception.

Dynemotion-Enabled NPC Brain Cycles

As is seen in the system architecture (Figure 1), Dynemotion is a complex system made up of many moving parts. To understand it more fully this section steps you through the cycle of a brain's think, act, and learn cycles. Typically these are done on a time schedule different for each cycle (1/4th of a second, 1 second, and 3 seconds). Knowing how brains work is imperative to incorporating Dynemotion successfully and creating convincing behavior scripts. The first and most rapid brain cycle is the think cycle. The think cycle typically happens every quarter of a second, but it is up to the simulation to determine this timing. Think cycles are kicked off by calling the DynemotionManager's Think() method. The time elapsed since the last think cycle is passed in as a parameter.

Think Cycle - during the think cycle a brain perceives all objects around it, and any incidents those objects are doing. It then calculates how it feels about the objects and incidents. The brain searches its own Associations for how it feels about objects, and executes the GetObservationEffect method of the behavior script that corresponds to the action of the incident, to understand how it feels about the incident. The feelings are all then applied to the brain's current target desire state. The brains current desire state and target are then compared to determine new desires based on max increases, max decreases and natural momentum. Finally the think cycle ends with all desires being modified (generally inhibited) by other desires as has been defined in the DynemotionSetup.xml file. For example, friendship doesn't matter much when a brain fears for its life.

Act Cycle - the act cycle is made up of two components, goal selection and action performance. Goal selection can be computationally expensive. In order to reduce this computational load, goal selection only occurs under certain scenarios: If a Brain has received a new Suggestion, if a Brain is finished performing its current action, or if the Brain's Desires have changed significantly (exceeding a threshold parameter) since the last Act cycle. The tolerance for change in Desires is set in the DynemotionSetup.xml file, see that section for more details. If none of these conditions is true, the Brain skips goal selection and executes its previously selected action.

Player characters also contain brains, but these brains do not perform goal selection at all; instead they check for a selected action coming from the player. If there is no selected action then the PC brain sets the selected action to its default move behavior (see behaviors xml) or default idle behavior, depending on whether or not the PC's body is currently in motion.

During goal selection an NPC brain goes through its short and long term incident memory, gathers actions proposed by observable objects, and applies existing suggestions to construct a series of goals. These goals are then ranked on the basis of how the brain thinks that goal's action will end up affecting its desires and to a lesser degree how well its current desire state matches the current desire state of the goal memory. Once the goals are scored suggestion modifiers are applied that could push goals lower or higher in the goal list. Then starting from the lowest scoring goal and working upward, each goal's IsPossible method (from its behavior script) is run to see if the action is currently possible. An action may be possible, impossible, or may indicate other actions that could be taken to make it possible. In this latter case, new actions (sub-goals of the first) are added to the goal list and are scored along with the rest. The first (lowest-scoring) action found to be possible, whether an initial goal or an action pursuant to another goal, is selected as the brain's current action.

Once a brain has decided on its action it checks to see if it has to cleanup an unfinished action that was previously in progress. If the selected action is not already in progress, the brain instantiates a new python behavior script by calling the corresponding behavior's _init_ method. Then the brain calls the

Execute method of the corresponding behavior script. Next an incident for this action is stored in the observation manager for observation by other brains. Finally the acting brain calls the GetSatisfaction method of the corresponding behavior script and applies that satisfaction to its target desire state.

Learn Cycle - during the learn cycle memories are processed. Any time a brain performs an action or sees another brain perform an action, the incident describing that action is captured and stored in the brain in short term memory. The brain then over time learns about that incident, in essence learning how that action made the performing brain feel. This allows it to later score the goal in its own attempts to determine how mimicking that action would make it feel.

Short term memory processing also applies the same learning process to all the Associations it has with the Terms in the memory. This can create new or modify existing Associations.

After being in short term memory for a period of time (which is settable by brain, brain type, or globally) the brain checks each memory for relevance. If it doesn't meet a certain (definable) desire threshold, then the memory is released (deleted). If it does meet the threshold then it is pushed into long term memory.

If a brain's long term memory already contains a memory with the same signature as the new one, the new memory's timestamps are added to the timestamp vectors of the existing long term memory. The desire effects are averaged in, weighted by the number of times this memory has been stored, using the timestamp vectors as a counter.

Cycle Timing - note that while Dynemotion Manager exposes the ability to call all brains to trigger a cycle, this is a brute force method. Storage of pointers to brains in your sim allows for you to call these cycles at your own pace. An example would be to split all the brains into four containers and run their act cycles independently. This would allow for all brains to not act at the same point in time.

Dynemotion API Contents (as found on the accompanying CD-ROM):
The Dynemotion API comes packaged as a lib file. It is then included in an application build using linker options. Then it is integrated into an existing simulation at the following levels:

- Simulation Interface (C++)
- Python Extension (C++)
- Dynemotion Manager (C++)
- Setup Files (XML)
- Behaviors (Python)

Work at all of these levels is required for Dynemotion integration, but once the initial three are completed most of the work for the last two bullets can be implemented by a designer with some limited technical experience. The Dynemotion API also comes with an example application, the Dynemotion Console. This shows examples of integrating Dynemotion into a simulation. In order to build the Dynemotion Console Visual Studio files you must create an OA_DYN_PATH environment variable that references the location in which you installed Dynemotion. PDF documentation for the API and integration is on the CD-ROM.

The Parametric Crowd Generator (PCG) application is also included on the CD-ROM. This is a prototype tool linking the creation of Dynemotion agents in a simulated world. It enables the user to choose a map and specify the parametric creation of groups of individuals who will be placed on that map. Each group shares a "brain type" as defined in Dynemotion, but each individual in a group may have different values for their personality and motivational desires. In addition, each is assigned a gender, age, and most importantly relationships with others in their own group and in other groups based on the parameters set by the user. Some of the aspects of this tool (e.g. map selection and group location on the map) are in a prototype state to demonstrate the essential functionality of creating heterogeneous groups based on a combination of pre-set brain types and group-based parameters.

Development Project Background Information

Online Alchemy was awarded this Phase II, Option I project on 2006JUL02 to continue in its development of software technology that would generate computer generated avatars or non-playable characters for potential use in military simulation and training applications. The contract award was based in part the history of accomplishments and prior development experiences of key management plus the ongoing Company research and development efforts in the area of commercial PC-based online video games.

This Option I project was based on Part 2 of the Statement of Work from the original Phase II technical proposal submitted on 2005JUL12 along with additional requirements and considerations specified by Total Immersion Software, Inc. the prime contractor on DoD's Real World project. The primary objective of this Option I award was to facilitate potential customer input (TIS) along with prior efforts in the Company's own commercial and military development experience in order to complete and release version 1.0 of the Dynemotion API for licensing and simulation integration development..

Features of the primary deliverable, the Dynemotion API version 1.0:

- provide a fully functional API that allows use of the Dynemotion Engine
- capability to modify individual NPC personality and goal/motivation parameters
- capability to modify groups and crowd definitions, relationships and goals/motivations (prototype crowd generation tool)
- model a scenario with a crowd inside the existing Online Alchemy simulation prototype generation of multiple NPCs with distinct (random or user-selected) personalities and behaviors

This project development and resulting software has assisted the Company in answering several technical questions, including:

- the ability to quickly adapt and integrate the API to multiple simulation engines
- performance constraints for running multiple life-like NPCs on COTS hardware in real-time
- ability to introduce a player-character into a simulation populated by NPCs
- novel new interface elements to enhance explanability of NPCs to users

Refer to pages 13-14 of this Report for additional details on "lessons learned" during the execution of this project and also for a list of new, unanswered or "open questions" that represent proposed additional research and development opportunities to explore the extent and potential of this technology.

The ability to create lifelike, autonomous computer characters, whether individually, in groups or in crowd simulations, is a continuing research and development priority for commercial, military and intelligence applications. In the military this need is reflected in Objective #4 of the DoD system-wide MS&T goals to "provide authoritative representations of human behavior for both individuals and groups" (DOD 5000.59-P Master Plan 1995). In the intelligence community this is reflected in requests to model personalities and behaviors in such a way as to simulate and predict reactions and responses to a variety of scenarios with variable objectives and inputs. In commercial video gaming the ability to create or generate autonomous, organic, sentient, responsive NPCs represents a significant opportunity for broadening market reach beyond existing "core" gamers to greater numbers of casual players and thereby increasing revenues.

In addition to addressing the above needs, such technology may create opportunities in the entertainment markets for licensing so that others may incorporate these advanced NPCs into third-party interactive entertainment and simulation products.

The Dynemotion technology appears to address several of these needs. Dynemotion combines an artificial intelligence model with an artificial psychology engine which may be used to create and generate believable characters in persistent world simulations. This will enable the creation of fully autonomous and contextually aware characters with individual personalities, emotions, goals, knowledge, values, standards, memories, and relationships that drive their actions. These NPCs interact with each other, their world, and eventually with human participants in psychologically and socially meaningful ways. The engine is based on recognized social science models and thus moves beyond primarily scripted behavior found in existing simulations to a richer set of possible actions, reactions and interactions which involve and are impacted by personality, knowledge, values, emotion, goals, and culture.

While there are several unique parts to the Dynemotion technology, the primary aspects that set it apart from others are the nuanced and evocative emotions experienced and displayed by the agents, the memory/knowledge architecture, and the learning and goal selection mechanisms used by the NPCs. These allow for more human-like interactions between agents (and eventually between the player participant(s) and agents), as well as for the natural representation of skills (procedural knowledge and experience), values (standards and normative relationships), and diverse relationships.

Potential Application and Impact for the DoD

Applications of this technology include persistent world games; military, educational, and training simulations; and psychological modeling and analysis tools. Using this engine simulation and scenario designers may create scenarios testing various assumptions and inputs that may be modeled, executed and measured in order to analyze and assess the life-like nature of NPC behaviors and outcomes. This ability to quickly simulate crowds and groups was a key objective communicated by TIS in development of the milestones for this Option I project and deliverables. The engine may be used to generate training simulation NPCs including local populations (townspeople complete with local knowledge, cultural norms, and relationships), various military personnel (e.g., an authoritarian, a respected colonel or a friendly, inexperienced lieutenant) and opponents with believable goals and behaviors (e.g., political rebels, religious extremists, terrorists, etc.).

Project Objectives, Scope & Methodology

This Phase II, Option I project focused on delivery of the initial version 1.0 API of the Dynemotion technology that also supports the Company's ongoing commercial development efforts and advanced the efforts of TIS' Real World project. The resulting design and development effort enabled the generation of autonomous, contextually-aware, believable NPCs that exhibit reasonable and expected behaviors and nuanced emotions, along with basic tools for parametric generation. The resulting technology enables the creation of NPCs that have unique personality and the developmental, goal-oriented, and relational context to provide plausible social and emotional responses to each other and to human simulation participants.

The final deliverable was to include simulated scene from TIS' Real World concept video trailer that involved the interaction between two military soldiers and an Iraqi crowd. The Company developed a scenario that was similar in look and feel to this video scene, however, it was populated by actual Dynemotion-enabled NPCs that reacted and responded to the context of the environment and characters around them, and not by pre-defined proximity or queue driven scripted AI. A Window Media Video of the crowd scenario was captured and sent to TIS in addition to the final version 1.0 of the Dynemotion API, since an executable license from the rendering software provider was not budgeted under the Company's prototyping license.

Benefits of the Project

The benefits of the Software design and deliverable include the potential for attaining Objective #4 of the DoD system-wide M&S to "provide authoritative representations of human behavior for both individuals and groups" (DOD 5000.59-P Master Plan 1995). NPCs generated by the Dynemotion engine may enable simulation training scenarios far beyond what is possible today. This technology may also lead to models/methods to assess the contextual believability and effectiveness of NPCs in general.

Dynemotion also has potential uses in a number of non-military applications including entertainment and non-military educational training simulations. Online Alchemy continues in its efforts to develop a massively multiplayer online gaming (MMOG) entertainment product that incorporates the Dynemotion enabled NPCs. It is also hoped that the API licensed by TIS will enable them to add believable, contextually-aware, realistic NPCs to their military simulation and commercial entertainment projects.

Non-military training and educational applications may include various law enforcement, emergency preparedness, crowd control and homeland security training scenarios ("across the seams" training). Finally, a variety of simulated individual or crowd interaction training simulations may be developed with Dynemotion, potentially facilitating knowledge transfer training in a variety of professional fields.

Project Innovation

The ability to create lifelike, autonomous, emotive NPCs individually and in crowds is likely to enhance a number of DARPA DARWARS, JSIMS, and USJFCOM simulation efforts as well as commercial projects.

This includes the introduction of new levels of realism in training scenarios such as:

- Squad-based information gathering and reconnaissance
- Crowd control, management and/or dispersal
- Potentially hostile enemy assessment, identification, and apprehension
- House to house weapons search and seizure
- Creation of culturally accurate computer-based characters, including enemy elements

The Dynemotion version 1.0 API presents a broad range of opportunities in both military and commercial spheres. Specific innovations represented in the Dynemotion version 1.0 API include:

- adaptation for MS&T of a novel, unique method of generating realistic, life-like NPCs
- novel and comprehensive integration of artificial intelligence and artificially psychology
- ability to evoke human emotional resonance from non-living computer generated characters
- multi-layered motivation-driven behavioral model with nuanced-based emotions
- human-like character memory model for managing NPC memories, which in turn affect behavior
- integration of emotive facial and body software components for realistic 3D representations

Online Alchemy was uniquely qualified to design, develop and deliver this software in a way that is technologically and commercially sound. This effort required an understanding of a diverse array of areas such as agent-based artificial intelligence, cognitive and emotional psychology, and development of large-scale online game systems. Online Alchemy's executives and engineering personnel possess unique experience and proficiencies which were required to develop and deliver this technology.

Using DARPA provided SBIR grants and internally generated funds, the company advanced the original design and development and of this innovative patent-pending software targeted for eventual use in its commercial multiplayer entertainment software. It is this design that was adapted, developed and delivered in fulfillment of this project's milestones.

Lessons Learned and Open Questions

During the course of the design and development stages of this research project, the Company has continued to learn a great deal about the nature of such a complex and innovative approach, as noted in previous reports on earlier related work for DARPA. During the execution of this project a number of new questions arose during this effort which may require additional research and development that was outside the scope of this Option I contract award. It is the hope of the Company to work with the Sponsor to pursue future projects and awards to further explore these open questions and additional potential of this technology.

Lessons learned during the execution of this project include:

- **3D GUI** – existing 2D graphical user interface ("GUI") elements were identified as insufficient in several areas required to adequately represent and communicate the complex and rich data made available in relation to Dynemotion-enabled NPCs. The Company began early evaluation of potential 3D UI tools and designs, and future projects might involve the creation of a "dashboard" for the NPC that would allow the end-user to modify the attitude, posture and the character's reflected & externalized emotional state. This may also provide additional opportunity for further research to address the area of implementation using a variety of novel 3D GUI indicators.

- **Explainability** – the importance and challenges (technical and GUI) of communicating what is being perceived, thought, considered and felt by Dynemotion-enabled NPCs. The project's Sponsor described to the Company the Ft. Benning orienteering training course, which is a rock by rock representation virtual simulation that has not provided much advantage over the real course by. Part of this is explained by the notion of "power line awareness" – the fact that there is a power line that runs down the actual trail which provides additional visual cues not available in the simulation. This begs the further questions of what surrogates are available in simulation to provide the needed realism and physical, emotion and other cues needed to adequately train and prepare today's war fighter. Surrogate issues like sounds, sun angle, etc. Additional research is merited to study this line of thought – the missing elements of proprioception not addressed, or addressable, in current simulations. This also includes the area that the Company deemed as a necessity to discover how to simulate emotional and social "power lines."

- There is much additional work that needs to be explored in the area of **advanced internal semantic model and** learning (e.g. widely discussed "bag of fertilizer" scenario and problem).
- **Reputation** – during the course of development the Company developed additional uses for the Dynemotion engine including the ability for NPCs to propagate trinitive opinion and externalized thought (w/o voter paradox or mono clustering). This technology might potentially be useful for the NIC geneogram and related projects. The Company has also received interest from third-party commercial interests for potential social networking and HR applications.

Open questions introduced by this project include:

- Continuing issues of implementation of greater flexibility in behavioral construction by the agents (known as 'flexible complements" internally). This along with more robust declarative memory structures and temporal associations will further enhance the realism and believability of the NPC agents
- Facial and body movements were advanced slightly in the current effort, but much research and development remains in order to overcome "uncanny valley" issues while still running on COTS.
- Stress testing still needs to be performed using the Dynemotion engine to support dozens and/or hundreds of NPC agents, and the minimal level of detailed AI (LODAI) implemented in version 1.0 of the API. These load tests on COTS systems will be required to better evaluate the long-term possibilities, potential trade-offs, and improvement priorities for follow-on revisions.
- The continuing challenges of developing new software engineering, development and QA processes to test and tune emergent processes and outcomes. This approach to scenario and simulation development represents a whole new field of study and research; proposed additions to the GUI feedback along with real-time telemetry-type measurement tools may provide additional ways to observe and test and collect real-time feedback from the NPCs.
- Integration of detailed culturally appropriate gestures and text to speech for NPCs remains an interest for possible integration, including tools like VCom 3D's Virtual Communicator software and other third-party natural language processing and procedural tools.
- Finally, there is a need to explore ongoing levels of realism and believability of the NPC actions, reactions and interactions over a longer period of time. Existing prototypes using the Dynemotion engine demonstrate that fidelity is maintained, but all scenarios and simulations delivered to date have been short-term in duration. Additional longer-term simulation of emergent behavior and impact of memory, association, values and learning are required to better understand and refine.

Anticipated additions post version 1.0 revisions and modifications (not specified under the current contract):

- Increased "level of detail" control including variable think/act cycle timing, multiple agents per brain, and behavioral level of detail
- Enhanced explainability: change-over-time in motivations, relationships, pursuant-action trade-offs, skill and value influences, etc.
- Additional brain types
- An increased library of stock available behaviors
- An increased library of motivational desires with greater emotional subtlety
- Increased performance optimization

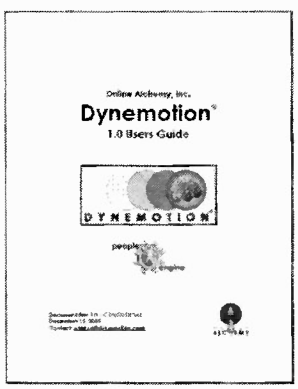
Research and Development Challenges
In the course of designing and developing the deliverables for this contract award it became clear to the Company that the nature of this project included several complex pioneering concepts and challenges. The process of creating self-directing NPCs that would behave in normally expected ways and yet possess enough behavioral choice and variety to maintain a semblance of independent thought and action was daunting, to say the least.

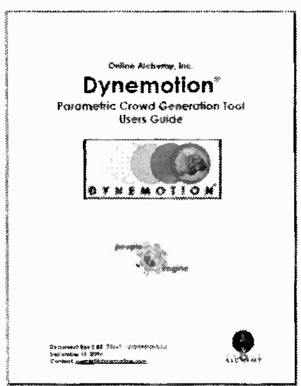
Summary of Project Deliverables
The following milestones outline Online Alchemy's development roadmap to the release of the Dynemotion version 1.0 API. These milestones were discussed and agreed upon by the Company and TIS, and frequent updates and uploads of work in progress were made by the Company. All deliveries of beta (pre-release) and the final version of the Dynemotion API were made on schedule and underwent methodical internal review, QA and testing. Development was performed using a modified Scrum Agile software development methodology (for more information reference http://en.wikipedia.org/wiki/Scrum_%28development%29)

At the end of Milestone #10 found on page 4 above (schedule for 2006DEC15), Dynemotion API 1.0 version was completed and delivered to TIS. This includes the ability for TIS and other technical customers to link the library to (C/C++) simulations and operate the complete cycles of creation, observation, action, and explanation, including the abilities to:

- Create or load multiple "brains" (to accompany intentional avatars or agents in the simulation) based on types provided by Online Alchemy or created by the licensing customer
- Create groups of parameterized brains (e.g. a crowd of civilians) with heterogeneous values
- Communicate observations of events, behaviors, objects, and people from simulation to the brains
- Control the thinking and act cycle timing
- Use behaviors provided, extend these, and/or write new behaviors
- Have the simulator field behavior requests from the brains based on internal goal-selection, execute them, and apply changes to the agents' brains
- Observe learning effects based on the brains' experiences
- Request explanations as to goals selected
- Save brains for later re-use

Based on internal testing on COTS hardware, and depending on other systems and simulation components that may be loading the CPU, a COTS PC should run several hundred Dynemotion brains simultaneously.


Online Alchemy, Inc.
Dynemotion
1.0 Users Guide


Online Alchemy, Inc.
Dynemotion
Parametric Crowd Generation Tool
Users Guide


Dynemotion Documentation

The above thumbnails are the cover pages of the electronic PDF documentation and Doxygen html index. They are located in the Documentation folder of the Dynemotion API v1.0 folder on the CD-ROM.

Development Milestone Details

Milestones #1-3 completed 2005SEP05

- Architectural integration and refactoring
- Initial API integration with sample simulator
- XML-based brain and desire definition
- Perceptual, relational, and behavioral memory structures
- Behavior scripting format, structures, and integration
- Emotional (disposition) communication with simulation
- Suggestions (goal strength modifiers)
- Austin Game Conference ("AGC") demonstration of early API to simulator integration

Milestones #4: Completed 2005SEP29

- Memory and performance optimization, post-AGC
- Initial stress and performance testing

- List of behavior scripts to be delivered in version 1.0 API
- Example behaviors, primarily for military simulations
- Product/architecture overview documentation
- Initial draft of usage documentation

Milestone #5-6: Completed 2005OCT13

- Improved logging and serialization
- Refine desire structures to include refactored external personality modifiers
- Revise memory structure (long term and short term); included differential learning times as defined per desire
- Extend Terms to relational thesaurus; enabled more flexible planning and conceptual/ontological comparisons
- Parameterized brain creation; included sample GUI to create groups using Gaussian randomized values within a specified range
- Initial behavior scripts and refactoring of initial human and animal scripts such as walk, run, greet, talk, threaten, hunt, attack, flee, soothe, etc. in support of TIS scenario demonstration deliverable
- Updated usage documentation

Milestones #7-8: Completed 2005NOV10

- Refine memory architecture, optimizing memory filtering and goal-culling
- Implemented declarative memory structures: non-behavioral knowledge, skill, and value memories
- Explainability, Phase 1: communicate numerical input for display in UI including goal strength, emotional impact, considered goals, operative suggestions
- Implemented Additional behavior scripts
- Development of Crowd Scene scenario based on Real World concept video trailer
- Updated usage documentation

Milestones #9-10: Completed 2005DEC15

- Complete action conditions, effects and flexible planning enabling much more flexible goal selection and explanatory capabilities; phrased in terms of declarative memory
- Explainability, Phase 2: text (assembled string) for display in UI including goal strength, pursuant conditions, significant Terms, considered goals, suggestions, skills, values
- Implemented additional behavior scripts
- Updated and finalized usage documentation in Adobe Acrobat PDF and html Doxygen formats
- Memory and performance optimization
- Completion and video capture of Crowd Scene scenario based on Real World concept video trailer
- Final QA and performance testing

Dynemotion v1.0 API Behaviors

Behaviors in Dynemotion are (generally short) scripts written in Python. They describe a modular action, its requirements, consequences, and observer effects. Behaviors may be as simple as Idle or Walk or as complex as Greet or Talk, which take into account the emotions, knowledge, values, and relationships of the individual speaking and their intended partner(s). Other related but more specific behaviors (e.g. Threaten and Persuade) can also be created.

While we anticipate that most technical customers will want to create their own scripted behaviors, we will include with Dynemotion a 'starter set' to show how these work and to give customers a jumping-off point in creating their own simulations.

Each behavior is scripted in as a class in Python and includes several functions:

- **__init__**: the standard Python constructor, used when an instance of a behavior is created
- **IsPossible**: a static/class function (requiring no constructed object) used primarily in goal selection that examines the agent considering this behavior and returns status indicating whether the action is possible, possible but missing certain conditions, or not possible.
- **Execute**: the primary function that is called whenever the agent performs an action. If an agent performs an action lasting more than one act cycle (this is typical), this function executes multiple times (once per act cycle) in a re-entrant fashion. This function returns a value denoting whether the behavior is completed or continuing, assuming the agent does not choose another action on the next cycle.
- **GetSatisfaction**: specifies the changes to the acting agent's motivational desire values based on their performance of this action. This may include logic to amplify or minimize some desire values based on the agent's success or other factors.
- **OnEnd**: A utility function used only when a behavior is complete or has been superseded by another behavior.
- **GetObservationEffects**: An important function that specifies changes in motivational desire, associations, etc., to anyone witnessing a behavior take place.

The following is the set of behaviors included in version 1.0 of the Dynemotion API. This list should not be considered to be entirely definitive, as other behaviors that exercise Dynemotion or support the TIS simulation scenario may have also been included in the final sample code and behavior set.

- Idle, Walk, Run, Flee, Patrol, Examine, Rest
- Greet, Talk, Farewell
- Threaten, Attack, Follow, Hunt, Defend, Cower
- Startle, Yell, Weep, Soothe, Laugh, Comfort

Please refer to the documentation on the CD-ROM for more detail on the behaviors included with Dynemotion API version 1.0.

Video Demonstrations (.wmv file format)

In addition to Dynemotion API version 1.0, sample code, behaviors and documentation, sample Windows Media Video of simulated scenarios developed as part of this project are included on the accompanying CD-ROM. Suggested system requirements to view these videos are: Intel® Pentium® processor with 1.5Ghz processor, Microsoft® Windows XP Professional or Home Edition, Windows Media Player, 512 MB of system RAM, CD-ROM drive, a 3D graphics accelerator card, multimedia sound card and stereo speakers and a color monitor with 1280 x 1024 resolution support, keyboard and mouse. To review, load the CD-ROM into the CD drive and navigate to the video demo folder.

Double-click on the file named "Dynemotion API Demos (narrated scenarios).wmv in order to view this video demonstration in Windows Media Player.

Commercialization Efforts
Online Alchemy's approach under this Phase II, Option I award was to continue ongoing development, testing and production of our primary commercial entertainment product while assessing the ability to transfer our existing development, knowledge base, experience and design to fulfill known MS&T goals.

Primary commercial development has been performed using entertainment and overall software design and development best practices, many of which have been initiated and demonstrated by the Company's founder, and known and published expert in multiplayer software design, development and deployment. Online Alchemy has met with potential investors to fund the full-scale production work required to complete the proposed commercial entertainment application by the target date of 2Q09. While the current investment environment for videogame entertainment software remains challenging, the Company hopes to secure additional investment in order to continue its implementation of the technology engine.

From the beginning the technology engine design has been created with dual purpose application at its core. The initial entertainment application is being developed not only as a revenue generating product, but also as the first instance of the Dynemotion engine, which is intended to develop additional revenues through sub-licensing of the engine to other entertainment and simulation developers. This is typically regarded in the software industry as "middleware" software licensing.

As noted earlier, the Company has realized early commercial success as a result of this Option I project as it was able to license the Dynemotion API version 1.0 to its first commercial licensee, which also happens to be the target end-user for this Phase II, Option I project, that is, Total Immersion Software, Inc. Additional inquiries have been received about licensing of the engine and the Company anticipates much future success in commercializing the engine and derivatives uses of the engine.

The Company continues to employ developers who are advancing these commercialization efforts.

Cost/Budget
As of the date of this Final Report the Phase II, Option I project expenditures were managed within the projected budget amounts, as proposed under this project. The Company secured additional contract help during the course of the project and was able to manage overall project costs and deliver on time and budget, as detailed above.

This contract has been executed to date at Online Alchemy's main office, 8000 Anderson Square Road, Suite 110, Austin, Texas 78757. This is a commercial grade office suite and includes staff offices, conference rooms and support facilities. To the best of our knowledge the property conforms to all local, state, and federal laws with respect to building codes, airborne emissions, waterborne effluents, external radiation levels, outdoor noise, solid bulk waste disposal practices, and handling and storage of toxic and hazardous materials. Existing computer equipment at Online Alchemy's facility includes top-of-the-line Pentium-class desktop development systems and a server supporting our distributed development environment. Platforms include C++ and AMD/Intel-based PC computer systems (Windows 95-98, 2000 and XP), and various specialized graphics and programming applications.

As of the date of this Final Report there are no additional SBIR projects underway or anticipated for the agency, although the Sponsor has discussed and alluded to possible future projects and uses for this technology. The Company remains open to the possibility of additional and future applications of their technology for related efforts within the DoD.

Project Outcome
As a deliverable of this project Online Alchemy has presented as part of this Final Report a Software CD-ROM which includes Dynemotion API version 1.0 and additional materials related to simulation integration and generation Dynemotion-enabled NPC. This software API and sample code may be used as a basis for integration and implementation in future DARPA MS&T applications. The resulting simulation integration may eventually be used for training evaluation, doctrine development, or possibly to stimulate current and future MS&T systems. Development of a believable parametrically based NPC generator or "People Engine" has the potential to make a significant contribution to the overall goal of continuously available, compelling, last-meter multiplayer training systems.

Summary

The challenge and opportunity that was addressed in this project was to complete development of the initial version 1.0 of the Dynemotion API in order to enable the simulation integration and generation of believable NPCs that may be used to simulate individuals and populations for a variety of military training scenarios and applications. The primary target application for this API was defined based on future project requirements of TIS's Real World simulation toolbox. Upon meeting with representatives of TIS post-delivery it appears that initial integration of the Dynemotion API version 1.0 will not take place until late 2007/early 2008, once the Real World simulation client is complete enough for API integration.

As part of the deliverable of the commercial license of this API to TIS, the Company has extended a one-year technical support offering that will commence once the Real World client software is ready for integration of this API.

Proposed Follow-on Research and Development

Online Alchemy desires to continue working with the Sponsor and the DoD to develop further enhancements and advancements to this software based upon the Dynemotion engine.

The proposed follow-on research and development work may result in the following outcomes:

- facilitate the development of a parametrically generated populations in existing or planned projects
- lead to the development of an enhanced "People Engine" tool and related measurement tools
- research and development related to exploring limits of NPC generation on COTS hardware
- research and development to enable online, distributed human player interaction with NPCs
- address various research and development "open questions" as highlighted in this final report
- compliance development so that the software may be used as part of other MS&T applications

The Company remains open to complementary efforts with existing or future SBIR award recipients for potential collaboration or cross-applications of discrete technologies to advance broader DoD objectives and initiatives. The Company is open to these types of discussions during the course of future development. While these additional technologies may not be required for the Company to complete a additional proposed projects, discussions may take place in support of the Sponsor's or DoD's overall objectives.

The Company would like to take this opportunity to thank the Sponsor, DARPA, DoD and the SBIR program for your support of our efforts. It has been a pleasure working with each of you, and we look forward to continuing efforts toward our mutual benefit.

Phase II, Option I Final report submission requirements checklist

- ☑ Final report for Phase II, Option I SB041-009 D2-0421 (W31P4Q-05-C-0286)
  - ☑ Summary of work accomplished per Phase II, Option I
  - ☑ Details of completion of the Statement of Work deliverables
  - ☑ Cost/Budget, facilities and personnel update
  - ☑ Company commercialization update
  - ☑ Proposed Follow-on Research and Development
- ☑ DD form 250 in support of this Final report (to AMCOM)
- ☑ DD form 882 in support of this Final report (to AMCOM)
- ☑ SF form 298 in support of this Final report (to AMCOM)

FINAL REPORT ABSTRACT (as required by 2005AUG30 Form 1423b-1 CDRL ATTACHMENT I):

ONLINE ALCHEMY HAS PERORMED RESEARCH ON THE APPLICATION OF ITS COMMERCIALLY DEVELOPED SOFTWARE TECHNOLOGY TO GENERATE AN API FOR COMPUTER GENERATED AVATARS OR NON-PLAYABLE CHARACTERS ('NPCS') FOR POTENTIAL USE IN MILITARY SIMULATION AND TRAINING ENVIRONMENTS. THE OUTCOME OF THIS RESEARCH AND DEVELOPMENT IS INTENDED TO PROVIDE SIGNIFICANT IMPROVEMENTS AND CAPABILITIES NOT CURRENTLY FOUND IN EXISTING NPCS AS THE STATE OF CURRENT TECHNOLOGY HAS LIMITED REALISM. THE MAIN DELIVERABLE OF THIS PROJECT IS THE DYNEMOTION API VERSION 1.0, AND RELATED SAMPLE CODE, BEHAVIORS, UTILITIES AND DOCUMENTATION THAT DEMONSTRATE THE ABILITY TO GENERATE NPCS WITH DISTINCT (RANDOM OR USER-SELECTED) PERSONALITIES, GOALS AND BEHAVIORS IN ORDER TO FURTHER DEMONSTRATE BELIEVABLE, PSYCHO-SOCIALLY PLAUSIBLE NPC INTERACTIONS, COMMUNICATIONS, EMOTIONAL RESPONSES, AND BEHAVIORS. THE PROJECT DELIVERABLES INCLUDES AN INTEL-PC, MICROSOFT WINDOWS XP OR XP PRO COMPATIBLE CD-ROM DATA DISC THAT INCLUDES THE AFOREMENTIONED SOFTWARE AND DOCUMENTATION DELIVERABLES.